Werner Kuhn <email adres>

Hitting the Complexity Barrier, Again

In the mid Eighties, a vendor used to brag about the number of commands in its system - 1800 as I remember. A complexity barrier had been hit, with users struggling to understand what a system could do for them, and how to find the right commands for their tasks. The vendor then proposed, in the early Nineties, a light version of its software that was much less complex, albeit less powerful. Couldn’t complexity be tackled without compromising functionality?

This question was a key motivation for the creation of the Open GIS Consortium in 1994. Addressing the complexity problem, OGC combined ideas from object-orientation (hiding complexity behind interfaces) and distributed processing technologies (allowing simple components to interoperate). The insights of deep thinkers in software engineering (such as Edsger Dijkstra, David Parnas, Joseph Goguen and others) had arrived into the geospatial technology arena.

Our industry’s own deep thinkers shared a belief that it was not only feasible, but ultimately profitable and even required by the market, to determine the right size of components and their "open sesame" interfaces, as Cliff Kottman called them. This belief was carried into executive rooms and software factories across the world. Finding the right levels of abstraction, wrapping data and functionality accordingly, and letting systems interoperate, began to produce promising results. Some of them were spectacular and refreshingly simple, such as the breakthrough web mapping demonstration in 1999.

The journey toward more versatile and more powerful software specifications and implementations turned out to be long and tedious. Some travelers started to become impatient and tired. Corporations and governments needed solutions now for their growing interoperability problems. Expensively collected data stores needed to be opened up to the public and to a supposed market of geodata buyers and resellers. Laws were mandating information access in a language reflecting the old stovepipe data exchange thinking.

Relief seemed to come with the idea of data interoperability: why bother to pass through narrow and slowly evolving service interfaces, if you can have direct access to data stores through a data exchange language? Unfortunately, this way, information hiding is going out the window and the meaning as well as the value of data become hard to define when data and operations get separated.

On top of these theoretical concerns, the “data interoperability” oxymoron and the effect of committee revisions of interface specifications have let the complexity monster raise its ugly head again. The latest manual of GML, the Geography Markup Language, weighs in at 373 pages and contains 34 schemata that need to be understood before working with the language. Specification teams spend weeks arguing whether easting or northing should come first in a coordinate tuple. Didn’t we once agree that it doesn’t matter, as long as there is an interface exposing each coordinate with clearly defined semantics drawn from a spatial reference system?

I am struck by the extent to which many people seem to accept or even like the new complexity. It must feel good to be an expert on complex technology. We seem to have lost sight of what we were after when we set out to tame the complexity monster ten years ago. Why is hardly anybody standing up and saying that the state of interoperability specifications is getting messy, beyond its justified evolutionary refinement? We even seem to forget that we knew how to cope with complexity: through simple interfaces to small and well-defined components. Ludwig Wittgenstein said that anything which can be expressed at all can be expressed clearly. Should we now conclude that geospatial information cannot be expressed at all?